

IWOMP 2020

Intel oneAPI Toolkits

Ronald W. Green, Manager Compiler Support



Agenda

- Intel® oneAPI Initiative
- Intel® oneAPI Toolkits
- Intel® Compilers in oneAPI Toolkits
- Using OpenMP* with Intel® Compilers
- Where to go to get Intel® oneAPI Compilers

oneAPI Industry Initiative

Alternative to Single-Vendor Solution

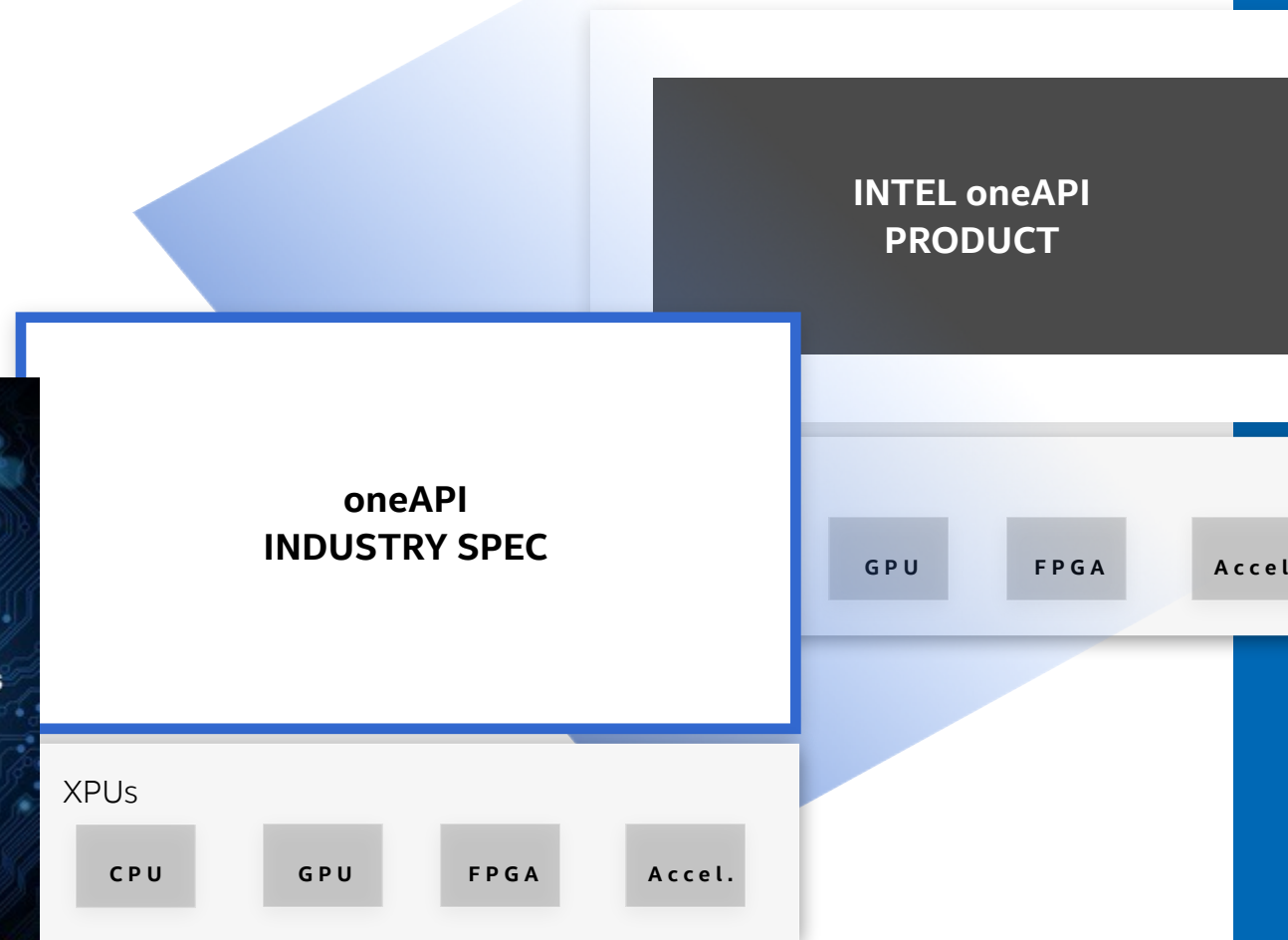


Open standard to promote
community and Industry support

Specifies Language, APIs, Low level
Hardware Interface

Enables code reuse across
architectures and vendors

Open call for comments at
[oneAPI.com](https://oneapi.com)



Intel® oneAPI Toolkits

TOOLKITS TAILORED TO YOUR NEEDS

Domain-specific sets of tools to get your job done quickly.



Intel® oneAPI Base Toolkit

A core set of high-performance tools for building Data Parallel C++ applications and oneAPI library based applications

[Learn More](#)



Intel® oneAPI HPC Toolkit

Everything HPC developers need to deliver fast C++, Fortran, & OpenMP* applications that scale

[Learn More](#)



Intel® oneAPI IoT Toolkit

Tools for building high-performing, efficient, reliable solutions that run at the network's edge

[Learn More](#)



Intel® oneAPI Rendering Toolkit

Powerful rendering libraries to create high-performance, high-fidelity visualization applications

[Learn More](#)



Intel® oneAPI DL Framework Developer Toolkit

Tools for developers & researchers who build deep learning frameworks or customize existing ones so applications run faster

[Learn More](#)

Toolkits Powered by oneAPI

Intel® System Bring-Up Toolkit

Tools to debug & tune power & performance in pre- & post-silicon development

[Learn More](#)

Intel® Distribution of OpenVINO™ Toolkit

Tools to build high performance deep learning inference & computer vision applications (production-level tool)

[Learn More](#)

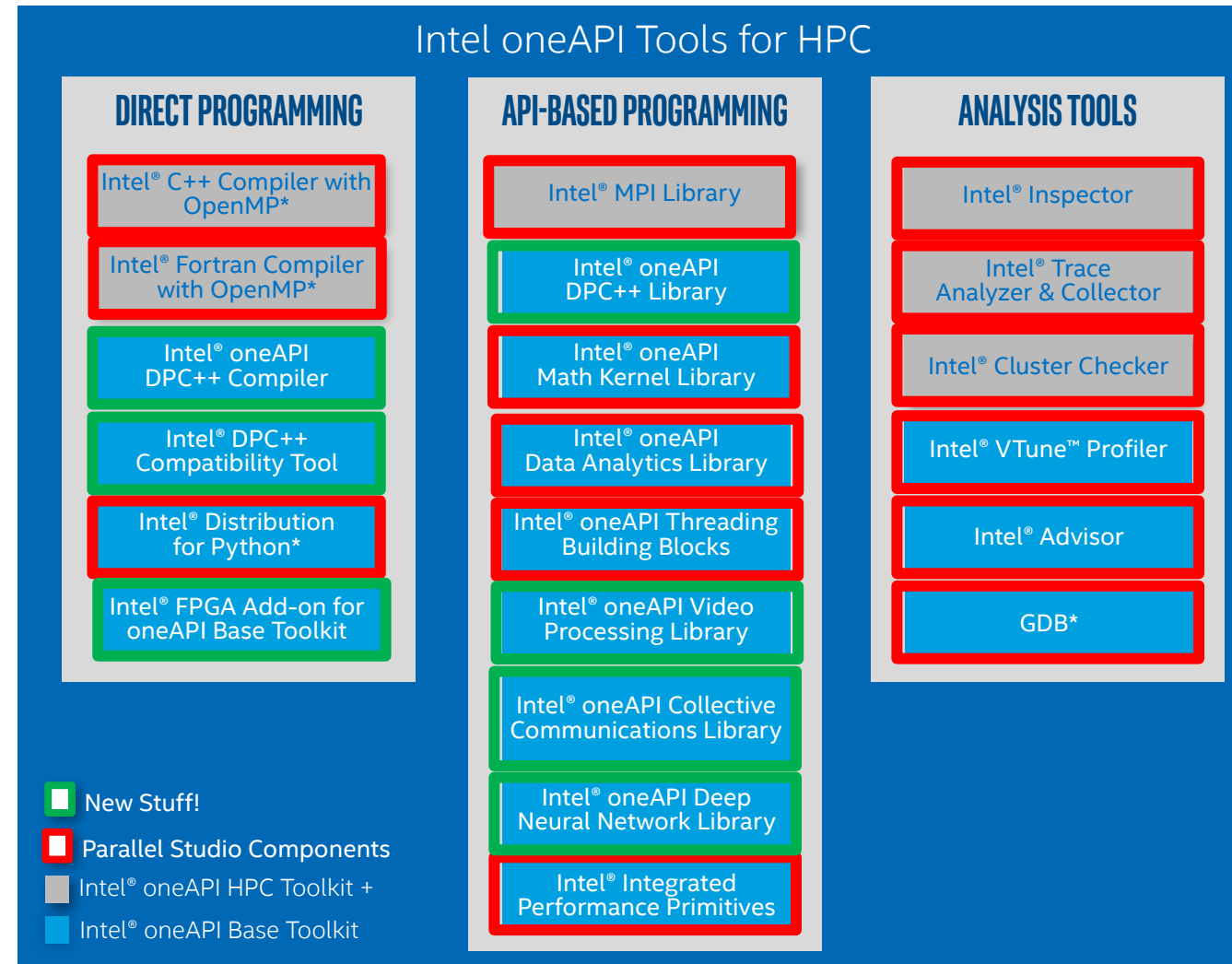
Intel® AI Analytics Toolkit

Tools to build applications that leverage machine learning & deep learning models

[Learn More](#)

Next Generation – The Same and MORE!

- All Parallel Studio Components today will be available in oneAPI Toolkits with *New Stuff!*
- Support continues without interruption
- *BETA today, Gold by end of Q4 2020*



Shopping for the RIGHT Compiler for You!

Intel® Compilers in Toolkits

Decoding your Intel® Compiler Choices



Key Concepts for Intel® Compilers Going Forward

- New Underlying Back End Compilation Technology based on LLVM
- New compiler technology available today in oneAPI Beta for DPC++, C++ and Fortran
- Existing Intel proprietary “ILO” (ICC, IFORT) Compilation Technology compilers provided alongside new compilers
 - CHOICE! Continuity!
- *BUT Offload (DPC++ or OpenMP TARGET) supported only with new LLVM-based compilers*

Packaging of Compilers

- PSXE 2020 for Continuity
 - Drivers: icc, icpc, ifort
 - v19.1 Compiler versions; 19.1 branch
- oneAPI Base Toolkit(BETA) *PLUS* oneAPI HPC Toolkit(BETA)
 - Existing IL0 compilers ICC, ICPC, IFORT in HPC Toolkit
 - v2021.1 code base; IL0 mainline code branch
 - **ADDED! New compilers based on LLVM* framework**
 - Drivers: icx, ifx and dpcpp
 - v2021.1 code base; new compilers mainline code branch (at this time)

Intel® Compilers – Target & Packaging

Intel Compiler	Target	OpenMP Support	OpenMP Offload Support	Current Status (Sep 2020)	Release Q4'20	Included in oneAPI Toolkit
Intel® oneAPI DPC++/C++ Compiler (dpcpp)	CPU, GPU, FPGA*	No	No	Beta	Production	Base
Intel® C++ Compiler, ILO (icc)	CPU	Yes	No	Production** + Beta	Production	HPC
Intel® oneAPI DPC++/C++ Compiler Pro (icx)	CPU, GPU*	Yes	Yes	Beta	Production	HPC
Intel® Fortran Compiler, ILO (ifort)	CPU	Yes	No	Production** + Beta	Production	HPC
Intel® Fortran Compiler (ifx)	CPU, GPU*	Yes	Yes	Beta	Production	HPC

Cross Compiler Binary Compatible and Linkable!

*Intel® Platforms

**PSXE 2020 Production, oneAPI HPC Toolkit(BETA)

Essential Takeaways for OpenMP Programmers

- The Intel® Compilers with OpenMP support are packaged in the Intel® oneAPI HPC Toolkit (BETA). BUT
- ICX (C/C++) needs Intel® oneAPI Threading Building Blocks which is in Intel® oneAPI Base Toolkit
 - And You probably want MKL, Vtune, Advisor : also in Base Toolkit
- AND ... For offload to Intel GPUs you need device drivers for your supported OS and hardware

<https://dgpu-docs.intel.com/installation-guides/index.html>

<https://software.intel.com/content/www/us/en/develop/articles/intel-oneapi-base-toolkit-system-requirements.html>

<https://software.intel.com/content/www/us/en/develop/articles/installation-guide-for-intel-oneapi-toolkits.html>

Let's Get Started!

Using OpenMP*

Intel® Compilers for OpenMP



OpenMP with Intel® Compilers

■ Drivers

- icx (C/C++) ifx (Fortran)

■ OPTIONS

-fopenmp

- Selects Intel Optimized OMP
- -fopenmp for Clang* O.S. OMP
- -qopenmp NO!! rejected, only in ICC/IFORT

-fopenmp-targets=spir64

- Needed for OMP Offload
- Generates SPIRV code fat binary for offload kernels

■ Setting up environment

source <inst dir>/setvars.sh

- Sets up env vars for all tools.
OR

```
source \  
<inst dir>/compiler/latest/env/vars.sh  
    "latest/" or specific version
```

- Other components also similar

```
<inst dir>/mkl/latest/env/vars.sh
```

Modulefiles also provided for each component in ../latest/modulefiles

```
<inst dir>/modulefile-setup.sh creates  
<inst dir>/modulefiles
```

Essential Intel env Var LIBOMPTARGET_PROFILE

- OpenMP Standard ENV vars are accepted. Add to this list ...
- `export LIBOMPTARGET_PROFILE=T`
 - performance profiling for tracking on GPU kernel start/complete time and data-transfer time.

```
GPU Performance (Gen9, export LIBOMPTARGET_PROFILE=T,usec)
```

```
... ..
```

```
Kernel Name:
```

```
__omp_offloading_811_29cbc383__ZN12BlackScholesIdE12execute_partEiii_1368
```

```
iteration #0 ...
```

```
calling validate ... ok
```

```
calling close ...
```

```
execution finished in 1134.914ms, total time 0.045min
```

```
passed
```

```
LIBOMPTARGET_PROFILE:
```

```
-- DATA-READ: 16585.256 usec
```

```
-- DATA-WRITE: 9980.499 usec
```

```
-- EXEC-__omp_offloading_811_29cbc383__ZN12BlackScholesIdE12execute_partEiii_1368:  
24048.503 usec
```

Debug RT env Var LIBOMPTARGET_DEBUG

■ Export LIBOMPTARGET_DEBUG=1

- Dumps offloading runtime debugging information. Its default value is 0 which indicates no offloading runtime debugging information dump.

```
./matmul
```

```
Libomptarget --> Loading RTLs...
```

```
Libomptarget --> Loading library 'libomptarget.rtl.nios2.so'...
```

```
Libomptarget --> Loading library 'libomptarget.rtl.x86_64.so'...
```

```
Libomptarget --> Successfully loaded library 'libomptarget.rtl.x86_64.so'!
```

```
Libomptarget --> Loading library 'libomptarget.rtl.opencl.so'...
```

```
Target OPENCL RTL --> Start initializing OpenCL
```

```
Target OPENCL RTL --> cl platform version is OpenCL 2.1 LINUX
```

```
Target OPENCL RTL --> Found 1 OpenCL devices
```

```
Target OPENCL RTL --> Device#0: Genuine Intel(R) CPU 0000 @ 3.00GHz
```

```
... AND MUCH MORE ...
```

For Latest Information

- Get Started with OpenMP* Offload Feature to GPU for the Intel® C++ Compiler and Intel® Fortran Compiler

<https://software.intel.com/content/www/us/en/develop/documentation/get-started-with-cpp-fortran-compiler-openmp/top.html>

Let's Get Started!

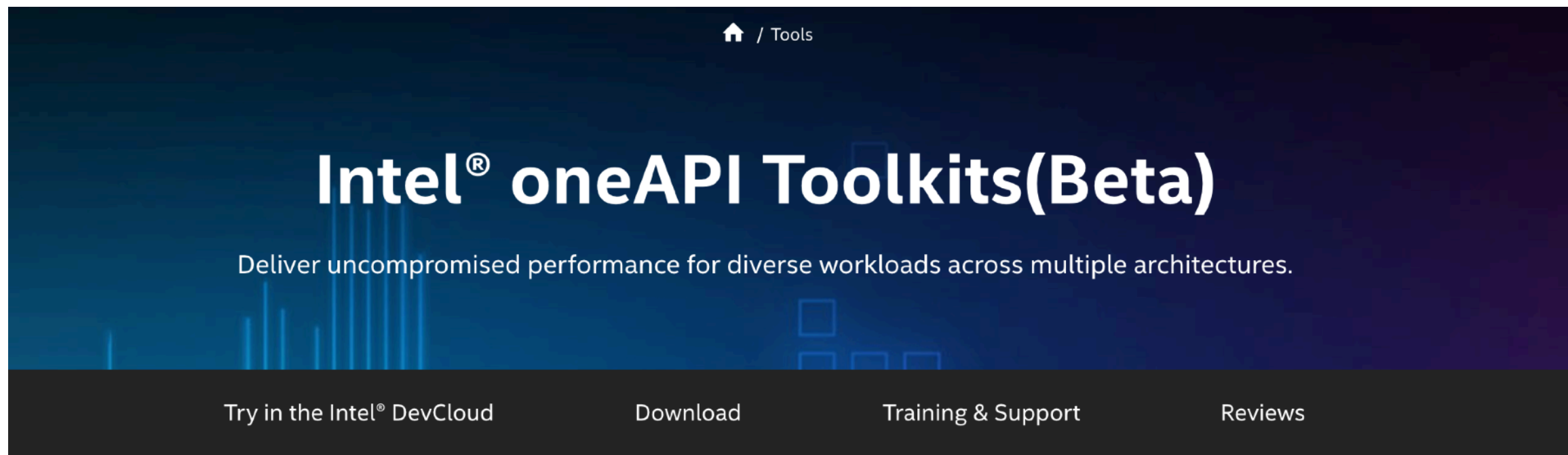
Getting Toolkits



Accessing oneAPI Toolkits/Components

- Intel® oneAPI Toolkits (Beta)
- Intel® DevCloud
- Online Learning Resources

Get the Intel® oneAPI Toolkits



Start here - <https://software.intel.com/en-us/oneapi/>

Expanded Component Accessibility and Acquisition

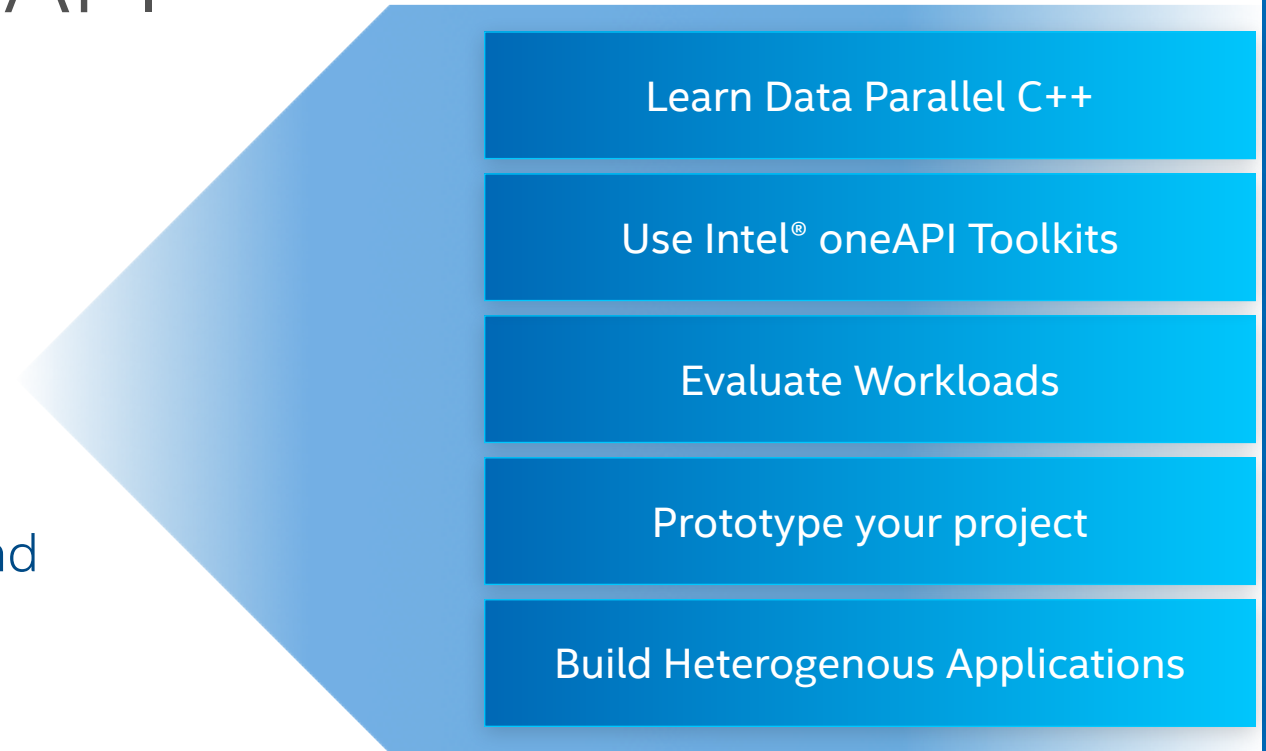
- Intel® oneAPI Toolkit Components at Gold launch will be available through:

- Intel® Registration Center
- NuGET
- APT & YUM Repositories (beta available today)
- Zypper (beta available today)
- Docker Containers (beta available today – Docker Hub)

<https://software.intel.com/content/www/us/en/develop/articles/oneapi-repo-instructions.html>

Intel® DevCloud for oneAPI

- A development sandbox to develop, test and run workloads across a range of Intel CPUs, GPUs, and FPGAs using Intel's oneAPI beta software
- For customers focused on data-centric workloads on a range of Intel CPUs, GPUs, and FPGAs



NO DOWNLOADS | NO HARDWARE ACQUISITION | NO INSTALLATION | NO SET-UP & CONFIGURATION

GET UP & RUNNING IN SECONDS!

software.intel.com/devcloud/oneapi

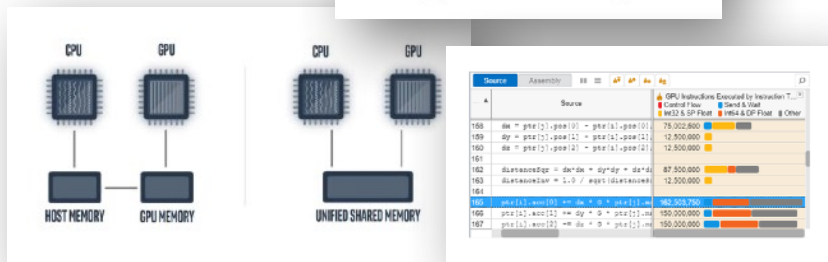
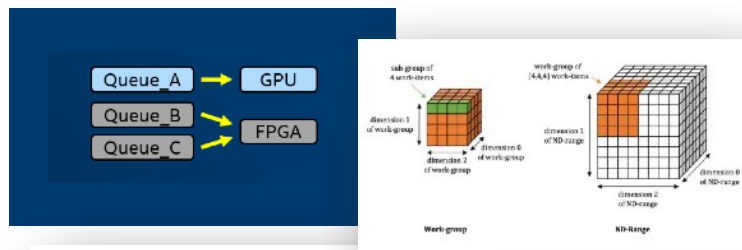
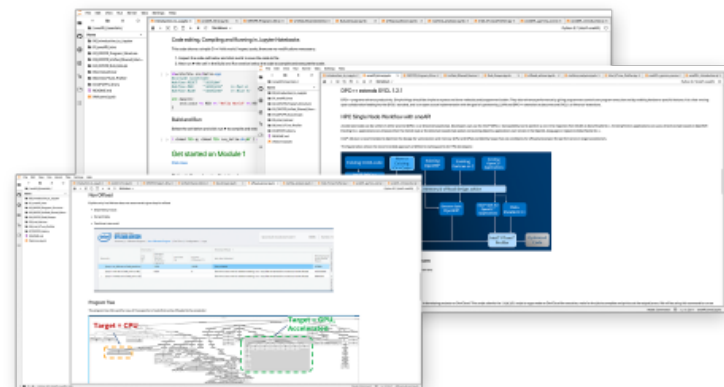
Essentials of Data Parallel C++

Start Learning DPC++

Get hands-on practice with code samples in Jupyter Notebooks running live on Intel DevCloud.

Learning Modules to Experience DPC++

- DPC++ Program Structures
- DPC++ Unified Shared Memory
- DPC++ Sub-Groups
- Demonstration of Intel® Advisor
- Intel® VTune™_Profiler



<https://software.intel.com/content/www/us/en/develop/tools/oneapi/training/dpc-essentials.html>

A close-up photograph of a person's hand wearing a blue nitrile glove, holding a square integrated circuit (CPU) chip. The chip has a green substrate and a dense array of gold-colored pins on its underside. The top surface of the chip shows various micro-components and a small label. The background is a blurred workshop or lab setting with various electronic components and tools.

QUESTIONS?

Notices & Disclaimers

- This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice.
- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at [intel.com](https://www.intel.com), or from the OEM or retailer.
- Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.
- INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS". NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.
- Copyright © 2020, Intel Corporation. All rights reserved. Intel, the Intel logo, Xeon, Core, VTune, and OpenVINO are trademarks of Intel Corporation or its subsidiaries in the U.S. and other countries. Khronos® is a registered trademark and SYCL is a trademark of the Khronos Group, Inc.

Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

